

# SPECIFICATION

Electronic Version 1.2.8

Stylesheet Version 1.0

## WEB BASED COMMUNICATION OF INFORMATION WITH RECONFIGURABLE FORMAT

### Background of Invention

[0001] The Internet is a very powerful tool for transferring and obtaining information. Many new applications have become available since the Internet's advent. Just to name a few, these include electronic placement of bids over the Internet, electronic determination of the user's bank balance over the Internet, automatic registration for various services over the Internet, electronic shopping with an electronic shopping cart, and various, virtually uncountable numbers, of others.

[0002] For all of the power of the Internet, it also has significant limitations. The language of the Internet, HTML in its various flavors, can only be properly interpreted by a device which is capable of reading HTML. Many different plug-ins and additional programs can be used, and are basically used as HTML windows. For example, the various flavors of Java, can enhance the Internet environment. Many companies have produced streaming video applications, such as shockwave and/or animation applications. These are basically shown as an associated application to the HTML.

[0003] Much of the customization which has been done on the Internet is done for the purpose of enabling more complexity, and more items to be shown over the Internet. However, browsing from a hand-held device has become more popular. For example, the Internet can be browsed from a cellular phone, from a PDA such as a palm VII, or other thin clients. Typically the way in which this is done is by forming a special web page for those thin clients, which runs a scaled-down flavor of HTML that is optimized for such thin clients. Only those special web pages which are optimized in this way can be effectively viewed. For example, any attempt to use such a system to browse a normal,

unoptimized web page, may cause an error, or may cause be information to be displayed in a way which makes it very difficult to effectively understand.

[0004] Other interactive hand-held devices, such as Blackberry devices and other devices which are optimized for e-mail and text messaging only, have no effective way to interact with the Internet. Therefore, while these devices are interactive in the sense that they can communicate and receive information via Internet-like channels, they can only receive information of a specified type.

## Summary of Invention

[0005] The present application teaches a system of using an Internet-aware service to form a conduit between the Internet and another device which may be a hand-held device or any other thin client, or even a regular client.

[0006] In one embodiment, a client which is capable of interactive communication, and more preferably a hand-held device, may be used to obtain specified kinds of information from the Internet. Requests are sent to a service. The service can be any computer which is capable of receiving requests, reformatting then into a specified form that is required by a web server, sending it, receiving the response, and reformatting that response.

[0007] The service will typically receive requests in some kind of e-mail format, e.g. plain text. The service may return the responses to the requests in either plain text or some flavor of XML. The service can its self be a web server or any other kind of service, and the service carries out translation in a way that enables obtaining information from the Internet, retranslating it, and sending the information to the hand-held device.

[0008] In another embodiment, the hand-held device may include the user's personal information, and which can be used to allow identification, e.g. login, to a specified Internet site.

[0009] Another embodiment describes effectively do-it-yourself Internet, where a service is used to rearrange the content and information in an Internet page, and to display the new information to the user.

## Brief Description of Drawings

[0010] These and other aspects will now be described in detail with reference to the

accompanying drawings, wherein: Figure 1 shows a diagram of flow between devices according to an embodiment; Figure 2 shows a flowchart of an embodiment to query an auction type website using a service; Figure 3 shows a flowchart of operation to query a bank; Figure 4 shows a flowchart of using this system with a general ecommerce site; Figure 5 shows a system of do it yourself formatting of internet content.

## Detailed Description

[0011] An overall flow of the communication between the various devices is shown in Figure 1. According to the system of Figure 1, a service is used to translate information between a hand-held device 100, and a web server 120. The web server 120 is connected to the Internet shown as 119, but more generally, can be connected to any publicly available source of information.

[0012] PDA/pager 100 can be any thin client which includes interactive capabilities. For example, it may be an e-mail pager, a cellular telephone, or any other device which is capable of interactively communicating. In one embodiment, the device may be limited to text messaging. The thin client 100 communicates as conventional over a channel 105 to a service provider shown generically as 110. The service provider can be, in the case of a cellular telephone, for example, a provider that operates as part of the cellular telephone system. The service provider also communicates with a service shown as 115. The service can be a conventional Web service or other subscription service, or simply can be an interfacing program that is running on a computer at the service provider's location. The service carries out certain translations, in order to communicate between the thin client, and the Web server 120. The service provider also includes a connection to the Internet, as conventional for such service providers, since they conventionally obtain information such as e-mail and Web content from the Internet.

[0013] While the embodiment describes the client 100 as being a "thin" client, it should be understood that this same flow can be carried out with other clients.

[0014] The operation of the system is shown in the flowchart of Figure 2. Figure 2 represents the flowchart for a system where the client 100 is an e-mail pager, and the system is being used to change or make a bid on an electronic auction site such as eBay. These same operations, however, would be carried out to exchange any kind of information with any kind of website.

[0015] At 200, the user makes the decision that they want to change or place an electronic bid. In order to start the process, at 200 the user sends an e-mail to a special address, here called simply "Web Service" with content in either or both of the subject line and/or the body of the e-mail indicating the website and what the user wants. The detail may be written in plain English, and in this embodiment is parsed by a keyword recognition system. More generally, the system may parse the plain English words using any kind of plain English recognition system.

[0016] The keyword recognition system which is used herein takes advantage of the fact that only certain allowable operations may be carried out in this way. For example, the e-mail may be sent to a special address that only accepts e-mails about operations to be carried out by the service. Only a limited number of operations may be carried out, for example, specified Web sites, specified banks, stocks, and others.

[0017] Therefore, when the e-mail says "I want to place a bid on eBay", then the system can parse different words from this e-mail. First, it can easily match the word "eBay", and once doing that, it knows that there are only certain things that can be done on eBay. For example, a user may be allowed to browse eBay categories, do a search on eBay by keyword, check on a previously made bid, or make a bid.

[0018] By looking at the remainder of the keywords, once recognizing "eBay", and noting the categories above (browse, search, check bid, make bid, up a bid where I've been outbid), the system sees the word "bid". This again narrows down the options; the allowable things the user can do are either check on a bid or make a bid.

[0019] The system may store a number of different ways of saying each one. For example, the system may store "place a bid", "make a bid", "bid on", to represent the make a bid option. The system may store "check (on) bid", "bid status", "how is my bid doing", for the other. The system then tries to match the words of the email, to the words that correspond to one of the allowed actions. The best correspondence is selected, if possible.

[0020]

Also stored within the database is a probability, for the specific user, of what they will want to do. This may be based on the user's previous actions. For example, the user may 77 percent of the time, want to place a bid when they use the word bid. The system may

also store global statistics of this type, in case the user's specific statistics are not available. These statistics may be used when the system cannot determine exactly what to do, as a basis for an educated guess.

[0021] For any of these, the system returns an e-mail saying "you want to place a bid on eBay, is this correct"? The user replies with yes or no to indicate whether the system has properly understood the command. If the user answers no, then the system may return a numbered list of things the user can do, based on its parsing. The user can then reply with a number, for example.

[0022] At 210, the thin client, here a pager, looks up personal information that is stored within the pager database. For example, a PDA or other pager may include a storage repository for information. More generally, the client may be any interactive-capable client that is capable of sending and receiving information, and displaying results to a user. The send and receive need not be in real time; that is; the receive may be delayed relative to the send by some amount.

[0023] This information may be either local, or stored on an on-line server such as "www.myPalm.com" or other. Alternatively, if personal information is not stored, then an additional interactive operation can be carried out similar to those disclosed below. In any case, this operation returns the user's name, login information (such as password) and other registration information necessary for the specified site, here eBay.

[0024] The pager will request eBay for a list of its current status. At 220, the service sends an e-mail to the eBay site, asking for stored information. Note that this step is done totally independent of the thin client. The service itself is interacting with the website, and receiving information that will be later reformatted in order to send to the thin client. The existing list of current status is received from eBay, and parsed into its different parts. For example, the favorites list, the items previously bid on list and the watch list may all be included. If the user, under the "content" item, has specified more detail about what they want, then this may narrow down what is actually retrieved, in which case possibly less than all of this information may be accessed.

[0025] Other commands of this type may be, for example, "I want to bid on an item that is on my watchlist". This may be parsed to retrieve the contents of the watchlist.

Alternatively, if the item says, "I want to bid on an item I've previously been outbid on", then the current bid list may be returned, or only items on the current bid list where the requesting party is not currently the winner.

[0026] Again, the information is requested by the service, and then received by the service. The service may parse and format this information in any desired way. Here, the service may parse the information at 225 into a text list, with each item having a special number that corresponds to a session ID. The number should preferably be a relatively short number, since the user will need to use this number to take further action on one of the items in the text list. In the embodiment shown, the session ID may be a01 referring to a first item, a02 referring to a second item, and so on. The actual content of the fields associated with item 1 may in fact be the description of the items, received from eBay. Similarly, other item numbers such as a03 may be provided. This list, again in text form, is sent as the body of an e-mail to the e-mail pager at 230. This embodiment describes using alphanumerics to describe the items, providing approximately 50<sup>3</sup> different possibilities. This may be far more than will ever be necessary, so long as the return address can be used to determine which session is being referred to.

[0027] Again, each item has a number. In order to select one of these numbers, the user sends an e-mail back to the Web service. This can be a reply to the previously-sent e-mail, or can be simply sent to an address related to numbers in the previously-sent e-mail. For example, the e-mail may be sent to the address IDnumber@Webservice.net, here this may be the specific ID number a01 being used. The web server looks at the return address, to determine the meaning of the session ID. With the return address, and the session ID, the web service can unambiguously determine the item to be bid on. From this, the web service determines the item at 245, and again sends an e-mail to the user requesting further details.

[0028] As part of the information received at 220, the Web service may receive existing bid amounts and other details necessary for placing the bid. The system might not have reported that information previously, but may still store it. Alternatively, this can be individually determined at 245.

[0029] Having gotten to this point, and the user selecting an item where a bid was made but now is no longer the highest bid, the system may conclude that the user intends to up

their bid. An e-mail is sent back to the pager at 250 that simply says "what new bid amount?" The user responds with a dollar number at 255 which is again sent to the Web service. The Web service again receives this number, which may or may not include the session ID number (here a01) in be subject line. This tells the Web service that the user wants to place a bid of \$xxx on the item number a01. The Web service now has all it needs it knows the item, the amount, and the user's personal identification information. An optional confirmation may be carried out at 260 where the Web service says "are you sure you want to place a bid of \$xxx on <item one>?", and waits for a yes or no by e-mail. If so, the system places the bid at 270 with eBay, and once receiving confirmation that the bid has been placed, reports that confirmation at 280.

[0030] On the eBay site, simply placing the bid does not mean that your bid is necessarily high enough. For example, the bid of xxx might not be higher then existing bids which already exist in the system. Therefore, the results of the bid, such as bid high enough or bid not high enough are reported at 280. This may be simply reported as "you are the current high bidder" or your bid has been placed but was not high enough.

[0031] The service may also carry out other functions once the bid is placed in this way. For example, the service may monitor eBay at increments of say 5 minutes, 15 minutes, or one hour, looking for outbid notifications. Since the operation is carried out with the server, other timed services may also be carried out. For example, the user may send the service a message saying "check the bid at every five minute intervals, and report to me. " The system can parse this out based on any real time is to parsing system, and decide what it's been told. The server can easily check the bid every five minutes, and send an e-mail to the users interactive device.

[0032] Again, since the service is running on a computer, any function that can be programmed in any language can also be carried out. Therefore, the message sent to the service can be much more complex. For example, the message can say "check the bid at every five minute intervals, and report to me only if the bid has changed." Any different combination of possibilities can also be programmed in this way. An advantage is that all of the user interface is carried out using plain text operations.

[0033] The above has described placing a bid on ebay, but it should be understood that this system can also be used for many other purposes, including buying virtually any product

on the Internet or others. Another example is for using this system to check on stock prices. An example command that could be sent to the service says "check on my usual stocks today, and report to me if any of them changed by more than 5 percent". A list of usual stocks can be stored in either the service or in the interactive device. The service can check on these for example at 10 minute intervals, and report by e-mail only those stocks which have changed. This shows the power of the format conversion system used by the present system. Rather than point and click, this system allows effectively plain language commands to be made.

[0034] An important advantage is allowed by the service that carries out the translation between the client and the actual web server that has the information. By providing this server, the client can access any source of information over any publicly available network, without actually having a device that can access the server.

[0035] Another embodiment is disclosed herein of using this system to obtain a bank balance over the Internet. In this embodiment, the operation is shown for use with either a text based device such as an e-mail pager, or with a device that is capable of Internet browsing.

[0036] At 300, an initial e-mail is sent to the Web service, the subject being bank balance. The system obtains the user's personal information at 305. This can be personal information that is already stored in the device, or entered information. At 306, the system determines if the information is stored locally, and if so flow continues. If not, the user is prompted to enter this information at 307.

[0037] At 310, the bank information is accessed, by the service accessing the web page associated with the bank. In order to obtain a bank balance, the user must send specialized information to the bank using their specialized web page. The web page may be stored, as a cached web page, and if so it is accessed at 315. If the web page is not stored at 315, the bank's website is called up at 320. In either case, the result is the web page, and the user's personal information such as account numbers, passwords and the like. The term "password" as used throughout this specification may include any personal identifying information including a password, a personal identification number, or some other identifying information.



[0038] At 325, the Web service arranges all of this, that is the personal information and the web page information, into a form of the specified type and sends this form to query the bank's website at 330. The bank's website once queried at 330, returns the desired information at 335. Presumably the desired information is on the first page returned, although the process can be continued multiple times if an additional page of information is necessary.

[0039] At 340 the received information is obtained in the HTML form and is reformatted into an XML form. A desired format for the XML form is applied at 345, which will be returned at 350. If the returning is to a text based device, control passes to the left in the flowchart, and text information is sent in a specified form. If the returning is to a browser type device, such as a PDA or a full-blown computer, control passes to the right in the flowchart, and the information may be sent in an XML form.

[0040] Information for both of the returned forms may be stored in the templates 370. Various forms of specified templates may be stored. For example, a template for a bank may be stored, for stocks, and also custom templates. Each of the templates describes the preferred format for the way in which information is returned. The template may be very simple; for example, for a bank the template may simply say "your balance is xxx, of which yyy is available balance". The variables in these templates may be filled in from the information obtained from the Internet site. The embodiment disclosed with reference to figure 5 shows how custom templates can be made.

[0041] Since this system obtains the raw information and formats it according to a template, a limited form Internet browsing device such as a PDA can effectively browser virtually any site, even those sites which are not available in reduced bandwidth/information versions. The effective or desired information on those sites is located. That information may be conveyed to the user as part of one of the forms/templates.

[0042] Another embodiment, shown with reference to Figure 4, is more optimized for use in a browsing type PDA, and can be used with the same features to carry out electronic shopping such as with a shopping basket on a site. Many of the general principles described above can be used. A generalized shopping form is shown as 400 includes the site name, which can be selectable from a list 402. The user selects the site name from the list, or types in their own desired site name. The user then types in the search term

and then actuates the execute key 406. This causes the PDA to form the information into an XML (or HTML) form at 410. The XML form is sent to the service at 415, which translates the form into a query in a format that emulates the user requesting specified information from the specified website. This query is sent to the specified website at 420. The website then returns results at 425. These results are again parsed into an XML form at 430, using one of the specified templates shown as 433. For purchasing an object, the template may be of the general form shown as 435 where one, two or four objects are located on the screen with reduced size pictures of each, the heading information for each, and the rest of the information for the object simply contained in an XML link marked <more> shown as link 436. If the user selects the "more" link, then additional information that was downloaded from the website is returned. Each returned page will similarly have a "more" link until reaching the last page at which only the "return" link is shown. The return link returns the user to the basic view 435. In any of the objects, main view, an "add to cart" icon 437 may also be viewed. In this embodiment, along the bottom of the view, a toolbar 438 allows selecting more results or changing the view. The changing the view for example may select a different format from the templates. Selecting this different format causes all further actions for the similar kind of site to be carried out using the new selected format.

[0043] The user can also signify checkout in the toolbar. When the user signifies checkout, the information from the service representing the simulated shopping cart is sent to the website, initiating the checkout process. The website may return conventional shopping cart information at 442, which is again parsed in an XML format at 444 into a list 446 that can be more easily displayed on the handheld. When the user signifies go, (448) all of the user's personal information is transmitted at 450 to the service, e.g. information already stored in the handheld. Alternatively, the user may be prompted for this information. The service sends the information to the website at 452, which confirms the buy. The confirmation is returned to the service at 454, converted to an XML form at 455, formatted and sent back as a confirmation 456.

[0044] Figure 5 shows yet another alternative system which is usable not only in thin clients but also in relatively more powerful clients. In this embodiment, a web page may be customized by a user. The web page customization software may effectively be a plug-in to the HTML reader that places a toolbar at the bottom. One item on this toolbar is a

customized link shown as 500. Upon executing the customized link 500, the computer that is running the software analyzes the HTML page and parses it into different parts. 510 shows how the different parts of the page may be parsed separately. The parsed portions may include for example fill in values such as 511, drawing parts such as 512, and fixed text such as 513 as well as plug-ins, e.g. Java or animations shown as 514. The system also brings out a blank page which may be located side-by-side with the parsed page, or may be located in a separate window. In this embodiment, each HTML object may be separately selected and moved. For example, if the variable 511 includes information such as a bank balance, the user can select this with their cursor, and drag it, as shown, to a desired area on the blank page. This makes an XML form which includes information from the other page. Any number of web pages can be combined together to form the different composite parts of the composite web page 520.

[0045] The composite web page may be continually updated. In order to obtain any of the various parts, the system may call up the original web page that produced the part, obtain the web page, take the portion of the web page corresponding to the selected part, and form the composite web page for eventual display. All of this may be done in the background. For example, the system may open all of these pages as real windows in the background, view the source, find the right part, import the right part, and then close the windows once the composite web page has been formed.

[0046] What this means, therefore, is that a number of windows may be dynamically opened, as background windows. Once the system has obtained the desired parts from those windows, they may be automatically close, with only the desired part being displayed on the composite page.

[0047] Effectively this provides do-it-yourself Internet, where parts of an HTML page are obtained, and those parts are used to form a composite form which is eventually displayed. In this embodiment, the composite form may be in XML, since that is easy to configure.

[0048] Although only a few embodiments have been disclosed in detail above, other modifications are possible. For example, the above has disclosed only certain limited kinds of thin clients. It should be understood that other clients, including personal computers, may be used.

[0049] The above has described the client being a machine such as a pager or the like. However, these same techniques may be carried out over a voice telephone type network. For the example, the user may enter their information either by touchtone keys, or by voice recognition. Systems of obtaining information from a telephone are well-known, such as bank by telephone type systems. Again, however, and analogously to the above, all of these require that be banks network have installed characteristics. In the present system, the telephone communication is with the equivalent of the service, who places a query to the bank or website. The website or bank must have some capability of being accessed. However, by using this service, the user can call, and using voice prompts and voice commands place a bid on eBay, obtain a bank balance, or anything else like that.

[0050] This system may also be used for obtaining many other kinds of information, basically too innumerable to specifically recite herein. For example, this system may be used to obtain current news stories, by sending the service a query "what are the top news stories?". Any kind of query of this type may be answered.

[0051] All such embodiments are intended to be encompassed within the following claims.